IN THE CLAIMS:

Claims 1-22 were originally filed with the instant application, and all of claims 1-22 are pending before entry of this document.

Claims 1-3, 7-10, 17, 19, and 21 are amended herein .

Claims 1-22 remain pending, and are reproduced here, with changes shown in redline:

1. (Currently Amended) A machine-readable medium having a data structure stored thereon for efficiently ordering a plurality of entities, ~~each~~ the entities~~y~~ having ~~a~~ respective ranks within a plurality of N ranks, the data structure comprising:

a horizontally linked list linking at least a subset of the plurality of entities in at least a descending rank order direction, ~~each~~ the entities~~y~~ in the horizontally linked list having ~~a~~ respective, unique ranks as compared to the ranks of other entities in the horizontally linked list; and,

an array having a plurality of fewer than N array entries over which the plurality of N ranks are distributed, such that ~~each~~ the fewer than N array entries~~y~~ ~~has~~ are associated with ~~a~~ respective, corresponding ranges of the N ranks, wherein the ranges of the N ranks represent subsets of the N ranks, at least one of the N array entries~~y~~ ~~each~~ pointing to an entity of the plurality of entities having a greatest rank within the ~~corresponding~~ range of ranks corresponding ~~for~~ to the at least one array entry.

2. (Currently Amended) The medium of claim 1, the data structure further comprising at least one vertically linked list, ~~each~~ the vertically linked list linking in at least one direction a corresponding subset of the plurality of entities having an identical rank.

3. (Currently Amended) The medium of claim 2, wherein ~~each~~ the vertically linked list links the corresponding subset of the plurality of entities in a first vertical direction and a second vertical direction.

4. (Original) The medium of claim 1, the data structure further comprising a head pointer pointing to an entity having a greatest rank of the plurality of ranks of the plurality of entities.

5. (Original) The medium of claim 1, wherein the horizontally linked list further links at least the subset of the plurality of entities in an ascending rank order direction.

6. (Original) The medium of claim 1, wherein the plurality of ranks are equally distributed over the plurality of array entries.

7. (Currently Amended) The medium of claim 1, wherein the entity having the greatest rank within the corresponding range of ranks for ~~each of~~ one or more of the at least one array entry is one of a subset of the plurality of

entities having the greatest rank within the corresponding range of ranks for the array entry.

8. (Currently Amended) The medium of claim 1, wherein at least one array entry of the plurality of array entries ~~each~~ points to null, corresponding to no entity within the plurality of entities having a rank within the corresponding range of ranks for the array entry.

9. (Currently Amended) The medium of claim 1, wherein ~~each~~ at least one entity of the plurality of entities is a thread, the rank of the entity is a priority for the thread, and the array is a priority queue.

10. (Currently Amended) A method implemented at least in part by a computing device for removing a particular entity from a plurality of entities as represented in a data structure for efficiently ordering the entities, ~~each~~ the entities~~y~~ having ~~a~~ respective ranks within a plurality of N ranks, the method comprising:

in response to determining that the particular entity is present within a vertically linked list linking in at least one direction a corresponding subset of the plurality of entities having an identical rank, the corresponding subset including the particular entity, delinking the particular entity from the vertically linked list;

in response to determining that the particular entity is present within a horizontally linked list linking at least a subset of the plurality of entities in at least in a descending rank order direction, the subset including the particular entity, delinking the particular entity from the horizontally linked list; and,

in response to determining that an array entry of a plurality of fewer than N array entries of an array over which the plurality of N ranks are distributed points to the particular entity, adjusting the array entry to point to one of null and another one of the plurality of entities.

11. (Original) The method of claim 10, wherein the array entry has a corresponding range of ranks, and adjusting the array entry to point to one of null and another one of the plurality of entities comprises, in response to determining that the particular entity was present within the vertically linked list, adjusting the array entry to point to a next entity within the vertically linked list.

12. (Original) The method of claim 11, wherein adjusting the array entry to point to one of null and another one of the plurality of entries further comprises, otherwise, in response to determining that the particular entity was present within the horizontally linked list, and that the rank of a next entity within the horizontally linked list is within the corresponding range of ranks for the array entry, adjusting the array entry to point to the next entity within the horizontally linked list.

13. (Original) The method of claim 12, wherein adjusting the array entry to point to one of null and another one of the plurality of entries further comprises, otherwise, adjusting the array entry to point to null.

14. (Original) The method of claim 10, further comprising, in response to determining that a head pointer pointing to an entity having a greatest rank of the plurality of ranks of the plurality of entities points to the particular entity, adjusting the head pointer to point to another one of the plurality of entities.

15. (Original) The method of claim 14, wherein adjusting the head pointer to point to another one of the plurality of entities comprises, in response to determining that the particular entity was present within the vertically linked list, adjusting the head pointer to point to a next entity with the vertically linked list.

16. (Original) The method of claim 15, wherein adjusting the head pointer to point to another one of the plurality of entities comprises, otherwise, in response to determining that the particular entity was present within the horizontally linked list, adjusting the head pointer to point to a next entity within the horizontally linked list.

17. (Currently Amended) The method of claim 10, wherein ~~each~~ at least one entity of the plurality of entities is a thread, the rank of the entity is a priority for the thread, and the array is a priority queue.

18. (Original) The method of claim 10, wherein the method is performed by execution of a computer program stored on a machine-readable medium by a processor.

19. (Currently Amended) A method implemented at least in part by a computing device for adding a new entity having a rank within a plurality of N ranks to a plurality of entities as represented in a data structure for efficiently ordering the entities, the entities also ~~each~~ having ~~a~~ respective ranks within the plurality of N ranks, the method comprising:

of a plurality of array entries of an array having fewer than N entries over which the plurality of N ranks are distributed, such that ~~each~~ the array entries~~y has a~~ correspond to respective ~~ing~~ ranges of ranks, determining ~~the~~ a particular array entry ~~having the~~ corresponding to a range of ranks in which the rank of the new entity lies;

adjusting the particular array entry having the corresponding range of ranks into which the rank of the new entity lies to point to the new entity in response to determining that the array entry currently points to null;

adjusting the array entry having the corresponding range of ranks into which the rank of the new entity lies to point to the new entity in response to determining that the array entry current points to an entity having a rank less than the rank of the new entity;

linking the new entity into a vertically linked list linking in at least one direction a corresponding subset of the plurality of entities having an identical rank, in response to determining that the rank of the new entity is equal to the rank of any other entity within the plurality of entities; and, otherwise, linking the new entity into a horizontally linked list linking at least a subset of the plurality of entities in at least a descending rank order direction, ~~each~~ the entities~~y~~ in the horizontally linked list having ~~a~~ unique ranks as compared to the ranks of other entities in the horizontally linked list.

20. (Original) The method of claim 19, further comprising adjusting a head pointer pointing to an entity having a greatest rank of the plurality of ranks of the plurality of entities to point to the new entity in response to determining that the rank of the new entity is greater than the rank of the entity of the plurality of entities to which the head pointer currently points.

21. (Currently Amended) The method of claim 19, wherein ~~each~~ at least one entity of the plurality of entities is a thread, the rank of the entity is a priority for the thread, and the array is a priority queue.

22. (Original) The method of claim 19, wherein the method is performed by execution of a computer program stored on a machine-readable medium by a processor.